EXPERIMENT EIGHT

🔬 Experiment : Constraint Satisfaction Problem

Implement a Constraint Satisfaction Problem (CSP) in Python for the following:

Part A – Without Using Constraint Module:

Find all (x, y) where x ∈ {1, 2, 3}, and 0 <= y < 10, and x + y >= 5.

```
# CSP Implementation Without Using Constraint Module

valid_pairs = []

for x in [1, 2, 3]:
    for y in range(10):
        if x + y >= 5:
            valid_pairs.append((x, y))

print("Valid (x, y) pairs where x + y >= 5:")
print(valid_pairs)
```

```
Valid (x, y) pairs where x + y >= 5:
[(1, 4), (1, 5), (1, 6), (1, 7), (1, 8), (1, 9), (2, 3), (2, 4), (2, 5), (2, 6), (2, 7), (2, 8), (2, 9), (3, 2), (3, 3), (3, 4), (3,
```

```
# Install the constraint module (only needs to be run once)
!pip install python-constraint

# CSP Implementation Using Constraint Module
from constraint import Problem

problem = Problem()

# Define the variables and their domains
problem.addVariable("x", [1, 2, 3])
problem.addVariable("y", list(range(10)))

# Define the constraint
problem.addConstraint(lambda x, y: x + y >= 5, ("x", "y"))

# Get solutions
solutions = problem.getSolutions()

print("Valid (x, y) pairs where x + y >= 5:")
print(solutions)
```

```
Collecting python-constraint
  Downloading python-constraint-1.4.0.tar.bz2 (18 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: python-constraint
  Building wheel for python-constraint (setup.py) ... done
  Created wheel for python-constraint: filename=python_constraint-1.4.0-py2.py3-none-any.whl size=24061 sha256=1f311ac934a23619159e6
  Stored in directory: /root/.cache/pip/wheels/1e/af/a9/990ae7e6a78319b7d7afaab2d14e7e5ae349d5613db5ae28fd
Successfully built python-constraint
Installing collected packages: python-constraint
Successfully installed python-constraint-1.4.0
Valid (x, y) pairs where x + y >= 5:
[{'x': 3, 'y': 9}, {'x': 3, 'y': 8}, {'x': 3, 'y': 7}, {'x': 3, 'y': 6}, {'x': 3, 'y': 5}, {'x': 3, 'y': 4}, {'x': 3, 'y': 3}, {'x'
```